IN THE CLAIMS

Please amend the claims as follows:

1. (Currently Amended) A method for encoding an image, comprising:

identifying adjacent blocks in the image;

identifying coding parameters for the adjacent blocks;

comparing the coding parameters between the adjacent blocks;

skipping deblock filtering ~~between~~ for removing image residuals caused by encoding the image according to the comparison of the coding parameters between the identified adjacent blocks when the comparison ~~coding parameters for the identified adjacent blocks~~ indicates that the adjacent blocks ~~are~~ have similar coding parameters; and

deblock filtering to remove image residuals between the identified adjacent blocks when the comparison between the coding parameters for the identified adjacent blocks indicates that the adjacent block do not have ~~are not~~ similar coding parameters.

2. (Original) A method according to claim 1 including skipping deblock filtering when the adjacent blocks have similar motion vectors pointing to a same reference image frame.

3. (Original) A method according to claim 1 including:

identifying transform coefficients for the adjacent blocks; and

skipping deblock filtering between the adjacent blocks when the transform coefficients are similar.

4. (Original) A method according to claim 3 including:

identifying D.C. components in the transform coefficients; and

skipping deblock filtering between the adjacent blocks when the D.C. components are the same or similar.

5. (Currently Amended) ~~A method according to claim 4 including:~~ A method for encoding an image, comprising:

identifying adjacent blocks in the image;

identifying coding parameters for the adjacent blocks;

skipping deblock filtering between the identified adjacent blocks when the coding parameters for the identified adjacent blocks are similar;

deblock filtering between the identified adjacent blocks when the coding parameters for the identified adjacent blocks are not similar;

identifying transform coefficients for the adjacent blocks;

skipping deblock filtering between the adjacent blocks when the transform coefficients are similar;

identifying D.C. components in the transform coefficients;

skipping deblock filtering between the adjacent blocks when the D.C. components are the same or similar;

identifying A.C. components in the transform coefficients; and

skipping deblock filtering between the adjacent blocks when the D.C. and A.C. components are the same or similar.

6. (Original) A method according to claim 3 including transforming the adjacent blocks using a Discrete Cosine Transform to generate the transform coefficients.

7. (Original) A method according to claim 1 including:

comparing blocks in the image with reference frames;

transforming the result of the comparison between the reference frames and the blocks in the image into transformed blocks having transform coefficients;

comparing the similarities between the transform coefficients; and

skipping deblock filtering between adjacent blocks in the image according to the results of the comparison between the transform coefficients.

8. (Original) A method according to claim 7 wherein the transform coefficients include D.C. transform coefficients.

9. (Original) A method according to claim 1 including controlling deblock filtering for a loop filter in an image coder.

10. (Original) A method according to claim 1 including controlling deblock filtering in one or both of a loop filter and a post filter in an image decoder.

11. (Original) A method according to claim 1 including:

identifying similarities between coding parameters in a luminance channel of the adjacent blocks; and

controlling deblock filtering for both the luminance channel and a chrominance channel in the image according to identified similarities in the luminance channel.

12. (Original) A method according to claim 1 including selectively skipping deblock filtering in any one of a H.261, H.263, H263+, MPEG-1, MPEG-2, or H26L encoding standard according to encoding parameter similarities between adjacent image blocks.

13. (Currently Amended) An encoder for encoding an image, comprising:

a processor adapted to identify adjacent blocks in the input image;

compare coding parameters ~~for~~ between the adjacent blocks; and

enable and disable filtering of blocking artifacts created by coding the blocks according to the coding parameter comparison between the adjacent blocks ~~according to the comparisons~~.

14. (Original) An encoder according to claim 13 wherein the processor is adapted to identify motion vectors and associated reference frames for the adjacent blocks and skip deblock filtering between the adjacent blocks according to the identified motion vectors and reference frames.

15. (Original) An encoder according to claim 13 wherein the processor is adapted to identify residual coefficients for the adjacent blocks and skip deblock filtering according to the identified residual coefficients.

16. (Original) An encoder according to claim 13 wherein the processor is adapted to skip deblock filtering according to D.C. components in the residual coefficients.

17. (Currently Amended) ~~An encoder according to claim 16 wherein the processor is adapted to~~ An encoder for encoding an image, comprising:

a processor adapted to identify adjacent blocks in the input image;

compare coding parameters for the adjacent blocks;

enable and disable filtering of blocking artifacts between the adjacent blocks

according to the comparisons;

skip deblock filtering according to D.C. components in the residual coefficients; and

skip deblock filtering according to D.C. and A.C. components in the residual coefficients.

18. (Currently Amended) ~~An encoder according to claim 13 wherein the processor is adapted to~~ An encoder for encoding an image, comprising:

a processor adapted to identify adjacent blocks in the input image;

compare coding parameters for the adjacent blocks;

enable and disable filtering of blocking artifacts between the adjacent blocks according to the comparisons;

compare blocks in the image with reference frames, transform the result of the comparison between the reference frames and the blocks into transformed blocks having transform coefficients, compare the transform coefficients and encoding parameters of adjacent blocks, and skip deblock filtering between adjacent blocks in the image according to the results of the comparison between the transform coefficients and encoding parameters.

19. (Original) An encoder according to claim 18 wherein the transform coefficients include D.C. transform components and the encoding parameters include motion vectors and reference frames.

20. (Currently Amended) A decoder for decoding an encoded image, comprising:

a processor adapted to identify adjacent blocks in the encoded image;

identify coding parameters for the adjacent blocks;

compare the identified coding parameters between the identified adjacent blocks; and

enable or disable filtering of blocking artifacts between the adjacent blocks according to the comparison of the coding parameters ~~for~~ between the adjacent blocks.

21. (Original) A decoder according to claim 20 wherein the processor is adapted to identify motion vectors and associated reference frames for the adjacent blocks and skip deblock filtering between the adjacent blocks according to the identified motion vectors and reference frames.

22. (Original) A decoder according to claim 20 wherein the processor is adapted to

identify residual coefficients for the adjacent blocks and skip deblock filtering between the adjacent blocks according to the identified residual coefficients.

23. (Original) A decoder according to claim 22 wherein the processor is adapted to identify D.C. components in the residual coefficients and skip deblock filtering between the adjacent blocks according to the identified D.C. components.

24. (Currently Amended) ~~A decoder according to claim 23 wherein the processor is adapted to~~ A decoder for decoding an encoded image, comprising:

a processor adapted to identify adjacent blocks in the encoded image;

identify coding parameters for the adjacent blocks;

enable or disable filtering of blocking artifacts between the adjacent blocks according to the coding parameters for the adjacent blocks;

identify residual coefficients for the adjacent blocks and skip deblock filtering between the adjacent blocks according to the identified residual coefficients;

identify D.C. components in the residual coefficients and skip deblock filtering between the adjacent blocks according to the identified D.C. components; and

identify similar A.C. components in the residual coefficients and skip deblock filtering between the identified adjacent according to the identified D.C. and A.C. components.

25. (Original) A decoder according to claim 20 wherein the processor is adapted to inverse transform the encoded image, compare blocks in the inverse transformed encoded image with reference frames, generate a reconstructed image from the comparison between the inverse transformed encoded image and the reference frame, and skip deblock filtering between adjacent blocks in the reconstructed image according to the coding parameters for the adjacent blocks.

26. (Original) A decoder according to claim 25 wherein the coding parameters include transform coefficients, motion vectors, and reference frame information.

27. (Original) A decoder according to claim 20 wherein the processor is adapted to skip deblock filtering in one or both of a loop filter and a post filter.